

Administração e Projeto de Redes

Tópico 3 – Camada de Aplicação Internet.

Prof. Carlos Augusto Latorre
prof_carlos_latorre@yahoo.com.br

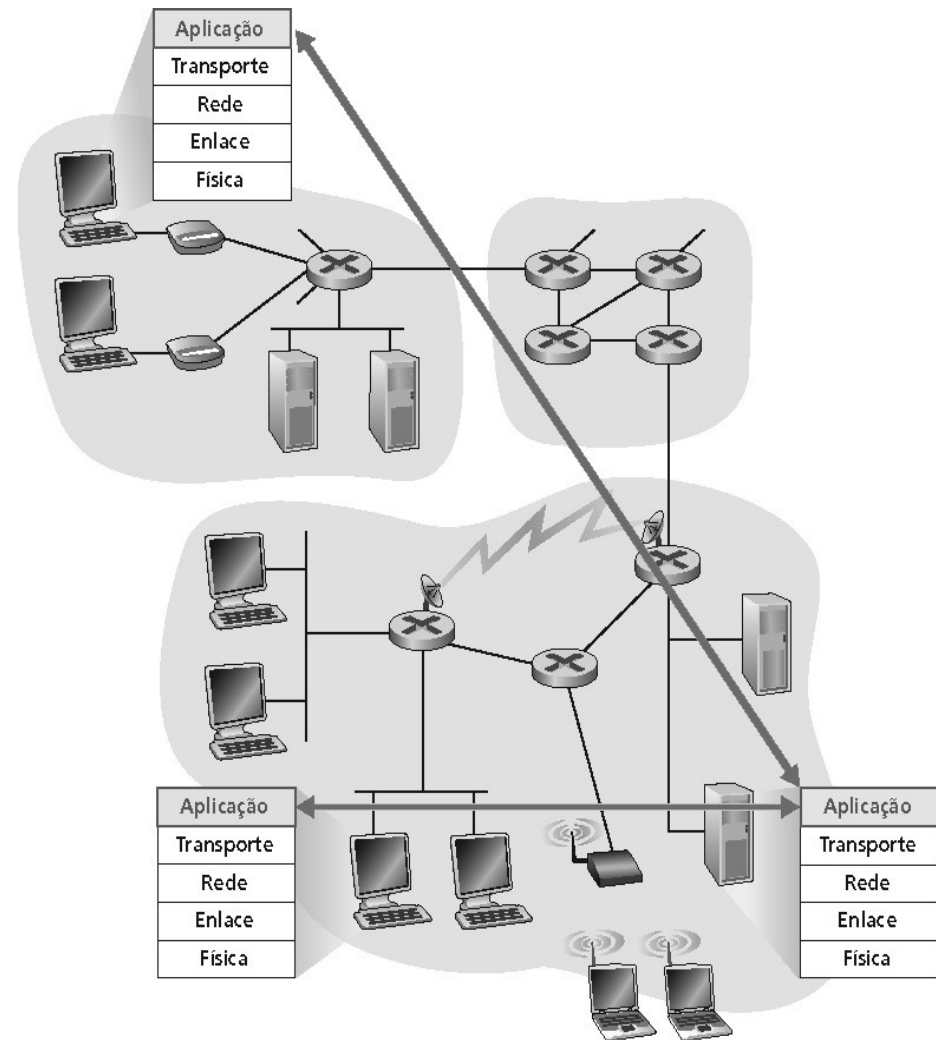
11/03/2010

Esclarecimentos

- Esse material é de apoio para as aulas da disciplina e não substitui a leitura da bibliografia básica.
- Os professores da disciplina irão focar alguns dos tópicos da bibliografia assim como poderão adicionar alguns detalhes não presentes na bibliografia, com base em suas experiências profissionais.
- O conteúdo de slides com o título “Comentário” seguido de um texto, se refere a comentários adicionais ao slide cujo texto indica e tem por objetivo incluir alguma informação adicional aos conteúdo do slide correspondente.
- Bibliografia básica: KUROSE, James F.; ROSS, Keith. Redes de Computadores e a INTERNET - Uma nova abordagem. Pearson. : , 2004.

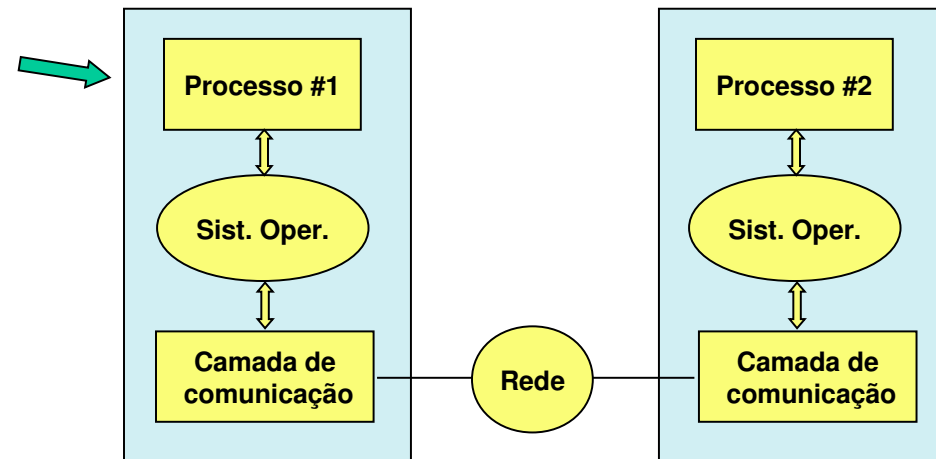
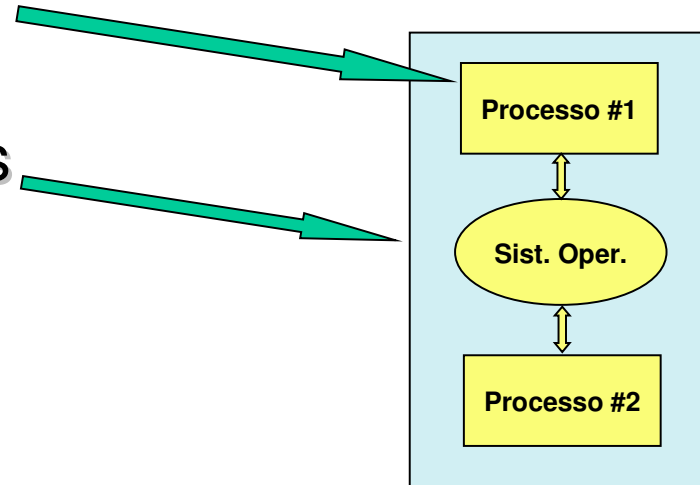
Criando uma aplicação de rede

- Escrever programas que serão executados em diferentes sistemas finais (hosts) de forma cooperativa e que se comunicam através de uma rede.
- Nenhum software é escrito para dispositivos do núcleo da rede (roteadores).
- Dispositivos do núcleo da rede não trabalham na camada de aplicação.



Comunicação de processos

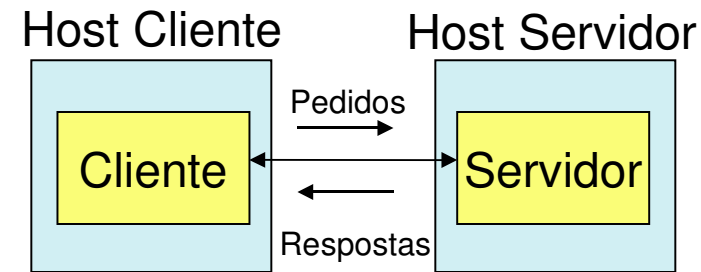
- Processo: é um programa que está sendo executado em um Host.
- Dentro do mesmo Host: dois processos se comunicam usando comunicação interprocesso (definido pelo Sistema Operacional).
- Processos em diferentes Hosts: se comunicam por meio de troca de mensagens usando os protocolos de comunicação existente entre os Hosts (exemplo: TCP/IP).



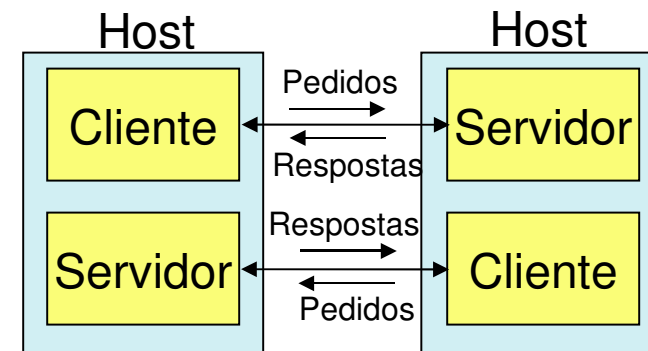
Aplicação Cliente/Servidor

- Aplicação distribuída:
 - Aplicação Cliente é processada no computador Cliente.
 - Aplicação Servidor é processada no Servidor.
- Módulo Cliente sempre inicia o processamento e solicita algo do módulo Servidor (Pedidos).
- Módulo Servidor recebe a solicitação do módulo Cliente e retorna uma resposta.
- A transação pode terminar no módulo Servidor ou no Cliente. Depende da solução definida pelo desenvolvedor do aplicativo.
- Aplicações com arquiteturas P2P possuem processos Cliente e processos Servidor que “conversam” entre si.

Cliente-Servidor

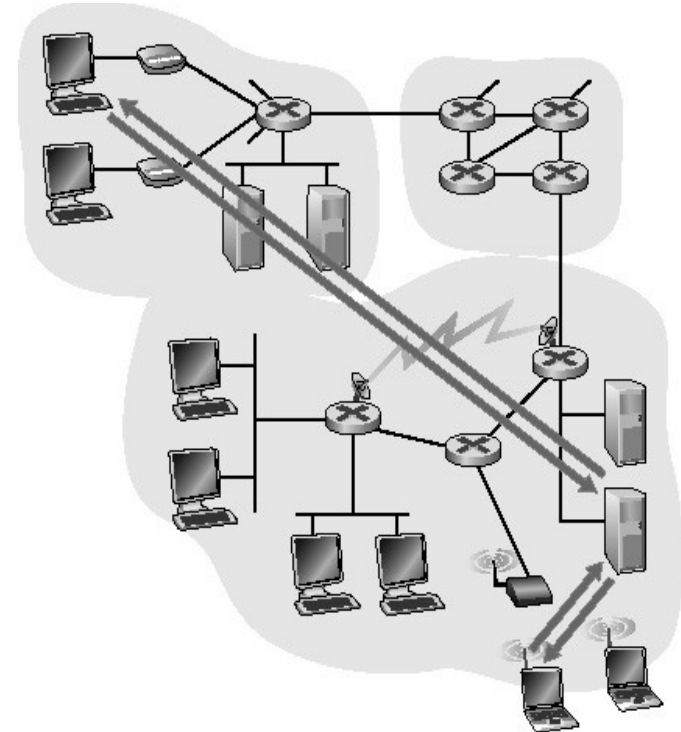


Peer-to-peer (P2P)



Características da Arquitetura Cliente-Servidor

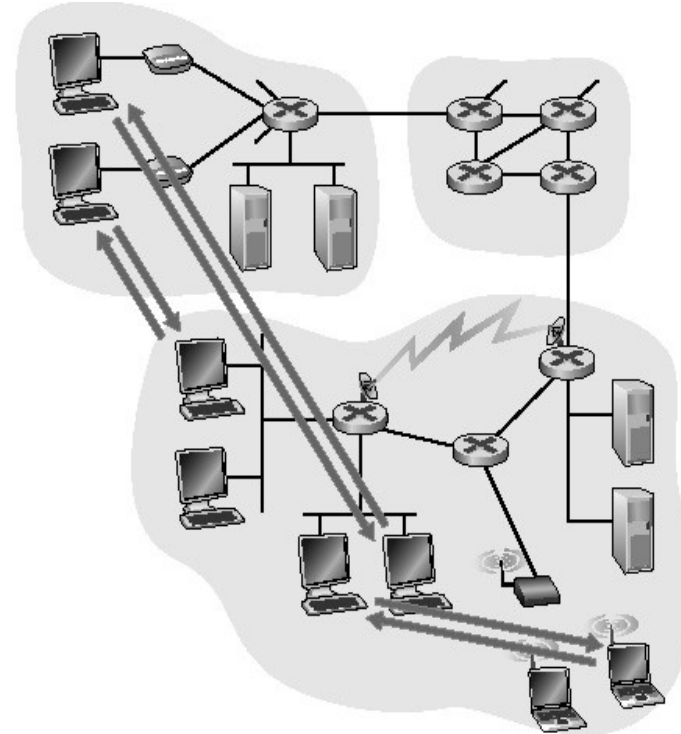
- Cliente:
 - Comunica-se com o Servidor.
 - Cliente não se comunica diretamente com outros Clientes.
 - Pode ser conectado intermitentemente.
 - Pode ter endereço IP e/ou Porta dinâmicos.
 - Solicita serviços ao Servidor.
 - Inicia a transação.
- Servidor:
 - Comunica-se com o Cliente.
 - Servidor não se comunica diretamente com os outros servidores.
 - Host sempre ativo.
 - Endereço IP fixo.
 - As aplicações usam Portas definidas.
 - Fornece serviços solicitados pelo Cliente.



a. Aplicação cliente-servidor

Características da Arquitetura P2P pura

- Sistemas finais arbitrários comunicam-se diretamente: aplicação Cliente de um host com a aplicação Servidor de outro host.
- Cada host tem uma aplicação Cliente e uma aplicação Servidor que se comunicam respectivamente com aplicação Servidor e Cliente de outro(s) hosts.
- Podem ter IP fixo ou dinâmico.
- Pares são intermitentemente conectados e trocam endereços IP entre si.
 - Ex.: Gnutella.
- Altamente escaláveis mas difíceis de gerenciar.



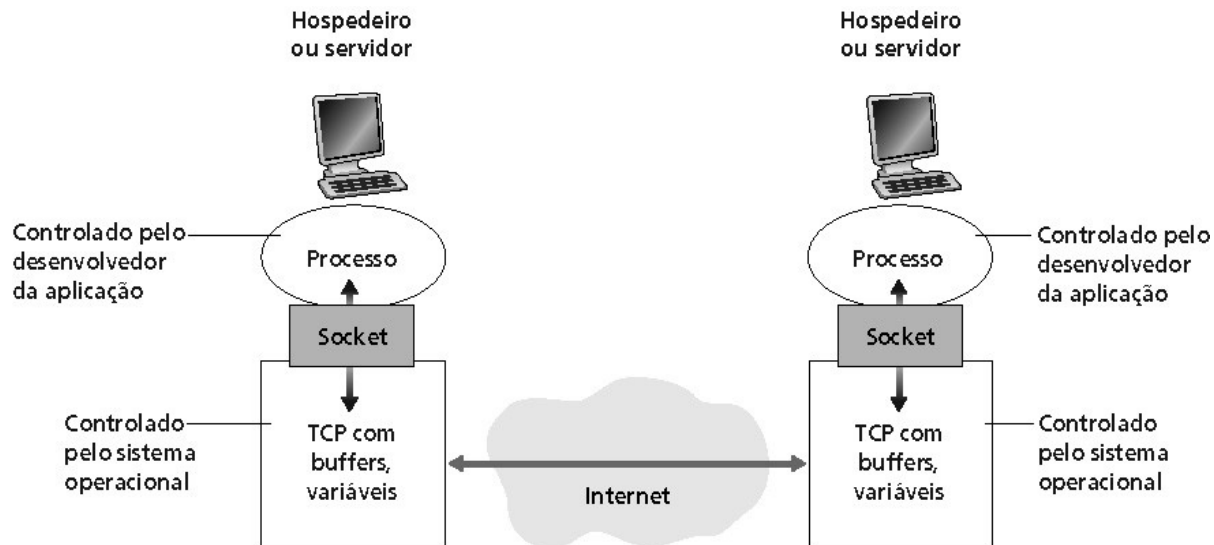
b. Aplicação P2P

Arquitetura Híbrida de Cliente-Servidor e P2P

- Solução de implementação que utiliza ambas estruturas (Cliente-Servidor e P2P) que são ativadas em momentos diferentes.
- Exemplo Napster: Transferência de arquivo P2P.
 - Hosts participantes são identificados em um Servidor Napster que mapeia os arquivos disponíveis para compartilhamento
 - Host Cliente deseja buscar um arquivo na rede e estabelece uma comunicação Cliente-Servidor com o Servidor Napster e identifica o arquivo desejado.
 - Servidor Napster localiza o host que tem o arquivo e indica o host destino ao solicitante.
 - Host Cliente se conecta ao Host Servidor (onde tem o arquivo desejado) e estabelece uma comunicação P2P.
- Exemplo Instant messaging (MS-Messenger):
 - Bate-papo entre dois usuários é P2P.
 - Detecção/localização centralizada de presença:
 - Usuário registra seu endereço IP no Servidor central do serviço quando fica on-line (comunicação Cliente-Servidor).
 - Usuário contata o Servidor central para encontrar endereços IP dos vizinhos ou de sua lista de contatos (comunicação Cliente-Servidor).
 - Ao solicitar conexão com o destino se estabelece uma comunicação P2P.

Sockets (“conecta” o processo da aplicação ao protocolo TCP)

- Processo que envia/recebe mensagens para/de seu socket (Processo de Transporte – Camada 4 do Modelo OSI).
- O socket é análogo a uma porta (por onde passam as informações trocadas pelos processos).
- O processo de envio empurra a mensagem para fora da porta.
- O processo de envio confia na infra-estrutura de transporte no outro lado da porta que leva a mensagem para o socket no processo de recepção.
- API (Application Programming Interface) : é um socket no mundo Microsoft Windows.



Processos de endereçamento de aplicações

- Para um processo receber mensagens, ele deve ter um identificador associado:
 - IP + Porta TCP.
- O endereço IP identifica o Host na rede.
- A Porta TCP identifica o processo dentro do Host (podem existir muitos processos e portas em uso simultâneo no Host).
- Exemplos de números de porta padrão de processo Servidor:
 - Servidor HTTP: 80.
 - Servidor de Correio: 25.
- www.iana.org (tem muitas informações sobre padrões).

O protocolo da camada de aplicação

- Tipo das mensagens trocadas:
 - Mensagem de requisição enviada pelo módulo Cliente.
 - Mensagem de resposta enviada pelo módulo Servidor.
- Sintaxe dos campos: são os campos que compõe a mensagem e como estão delineados no pacote da mensagem. Define o formato da informação na mensagem. Por ex: campo numérico com 4 posições.
- Semântica dos campos: é o significado da informação dos campos da mensagem. Por ex: campo Ano de Nascimento (do exemplo acima).
- Regras que definem quando e como os processos enviam e respondem as mensagens (protocolo).
- Protocolos de domínio público:
 - Definidos nas RFCs.
 - Recomendados para interoperabilidade (Ex.: HTTP, SMTP).
 - Protocolos proprietários (Ex: KaZaA).

Esclarecimentos TCP (Transmission control protocol) – Aplicações

- O TCP é um protocolo orientado a conexão e é dito confiável porque garante a entrega de todos os pacotes transmitidos pelas aplicações, recuperando eventuais pacotes perdidos e entregando todos na mesma sequência transmitida.
- Uma conexão lógica é estabelecida previamente entre os hosts, possibilitando a administração do processo de transferência dos pacotes. Identifica os pacotes recebidos pelo destinatário e retransmite-os em caso de perda de pacotes.
- O TCP oferece um método bastante confiável para transmitir cadeias de dados em formatos de byte entre aplicações que usam protocolos como HTTP, FTP, TELNET, POP3, SMTP, entre outros, isso é, as transações precisam de troca de pacotes entre os hosts dentro da mesma sessão de trabalho.

TCP – Aplicações

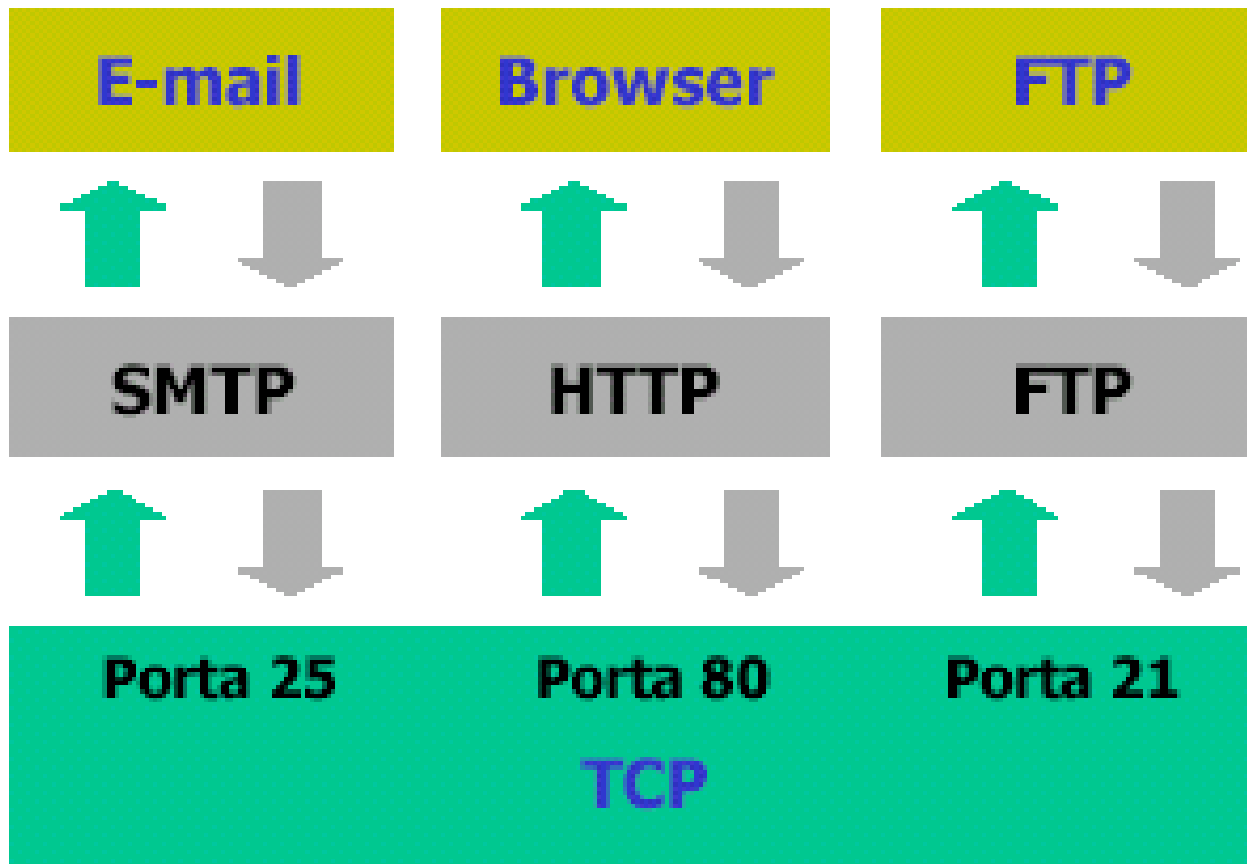
- HTTP:
 - HyperText Transfer Protocol (Protocolo de Transferência de Hipertexto): é o protocolo usado para a transmissão de dados no sistema World-Wide Web (WWW). Cada vez que você aciona um link, seu browser realiza uma comunicação com um servidor da Web através deste protocolo.
- FTP:
 - File Transfer Protocol (Protocolo de Transferência de Arquivos): é o meio de transferência de arquivos de diversos formatos, sendo desde uma foto, um texto, um artigo científico, uma página de Internet, um programa, um MP3, de um host a outro.

TCP – Aplicações

- TELNET:
 - Terminal Virtual: é o protocolo Internet para estabelecer a conexão entre computadores. Através dessa conexão remota, pode-se executar programas e comandos em outra máquina, como se o teclado de seu computador estivesse ligado diretamente a ela (simula remotamente um Terminal Local).
- POP3:
 - Post Office Protocol (POP3) é um protocolo utilizado no acesso remoto a uma caixa de correio eletrônico no Servidor de E-mail. Permite que todas as mensagens contidas numa caixa de correio eletrônico possam ser transferidas seqüencialmente para um computador local. Então o usuário pode administrar seus emails em seu computador local (ler as mensagens recebidas, apagá-las, responder-lhes, armazená-las, etc.).
- SMTP:
 - Simple Mail Transfer Protocol (SMTP): é o padrão de fato para envio de e-mail através da Internet. O SMTP é um protocolo relativamente simples, baseado em texto simples, em que um ou vários destinatários de uma mensagem são especificados (e na maioria dos casos validados), sendo depois a mensagem transferida de um Servidor de email a outro(s).

Portas Padrão no Protocolo TCP (exemplos)

- Combinação de Endereço IP e Porta = Socket.



UDP - Aplicações

- User Data Protocol (UDP) é um protocolo não orientado à conexão, não confiável e portanto com pouco “overhead” (sobrecarga). É usado por protocolos da camada de aplicação como DNS, ICMP, SNMP e TFTP, entre outros.
- Usado para transmissão de dados em tempo real, protocolos de gerenciamento (SNMP) e resolução de nomes (DNS).
- A vantagem para programas usarem o UDP em vez do TCP é o tempo de resposta. Primeiro porque o tamanho do pacote UDP é menor, já que o cabeçalho UDP é bem menor que o TCP, e segundo, no UDP não existe um mecanismo de verificação de chegada do pacote, acelerando o envio da mensagem.
- O processo de garantia de entrega e validação dos dados transmitidos deve ser tratado na camada de aplicação, se necessário.

UDP - Aplicações

- DNS:
 - Domain Name System: Executa o serviço de tradução de Nomes nos endereços IP correspondentes aos servidores onde esses nomes estão associados. As aplicações não precisam conhecer o endereço IP do host destino, bastando ter o Nome e conhecer o IP do Servidor DNS que fará a tradução do Nome pelo IP do Servidor Destino.
- ICMP:
 - Internet Control Message Protocol: é um protocolo integrante do Protocolo IP, utilizado para funções operacionais da rede como controle de erros de transmissão e fornecer relatórios de erros à fonte original. Qualquer computador que utilize IP precisa aceitar as mensagens ICMP, e alterar o seu comportamento em resposta ao erro relatado.
- SNMP:
 - Simple Network Management Protocol (Protocolo de Gerência Simples de Rede) é um protocolo usado para gerência operacional de redes TCP/IP. Permite a supervisão e controle remoto dos componentes da rede TCP/IP.
- TFTP:
 - Trivial File Transfer Protocol: é um sistema muito simples de transmissão de arquivos pela Internet. Não inclui sistemas de autenticação e é usado apenas para o boot remoto de sistemas.

Como as características de desempenho da rede impactam o desempenho das aplicações?

- Perda de dados durante a transmissão de dados na rede:
 - Algumas aplicações (ex.: áudio) podem tolerar alguma perda (sugestão: usar UDP).
 - Algumas aplicações (ex: transferência de arquivos, Telnet) exigem transferência de dados 100% confiável (sugestão: usar TCP).
- Temporização (atraso, latência).
 - Algumas aplicações (ex.: telefonia Internet, jogos interativos) exigem baixa latência (sugestão: usar QoS – Qualidade de Serviço = priorização no transporte de pacotes de dados na rede).
- Banda passante:
 - Algumas aplicações (ex.: multimídia) exigem uma banda mínima para serem “efetivas” (sugestão: dimensionar adequadamente o link e a rede).
 - Algumas aplicações melhoram o desempenho do tempo de resposta quando a banda disponível “aumenta” (sugestão: usar banda por demanda - “on-demand”).

Comentário: Aplicações comuns e requisitos para o serviço de transporte

Aplicação	Perdas	Banda	Sensível ao atraso
file transfer	Sem perdas	Elástica	Não
e-mail	Sem perdas	Elástica	Não
Web documents	Tolerante	Elástica	Não
real-time áudio/vídeo	Tolerante	áudio: 5 Kb-1 Mb vídeo:10 Kb-5 Mb	Sim, décimos de segundo
stored áudio/video	Tolerante	áudio: 5 Kb-1 Mb vídeo:10 Kb-5 Mb	Sim, segundos
jogos interativos	Tolerante	kbps	Sim, décimos de segundo
e-business	Sem perda	elástica	Sim

Visão geral do HTTP (1/2)

- HTTP - HyperText Transfer Protocol.
 - Protocolo da camada de aplicação da Web.
- Modelo Cliente/Servidor:
 - Cliente: browser (programa no micro do usuário) que solicita, recebe e apresenta objetos da Web.
 - Servidor: envia objetos em resposta a pedidos do browser.

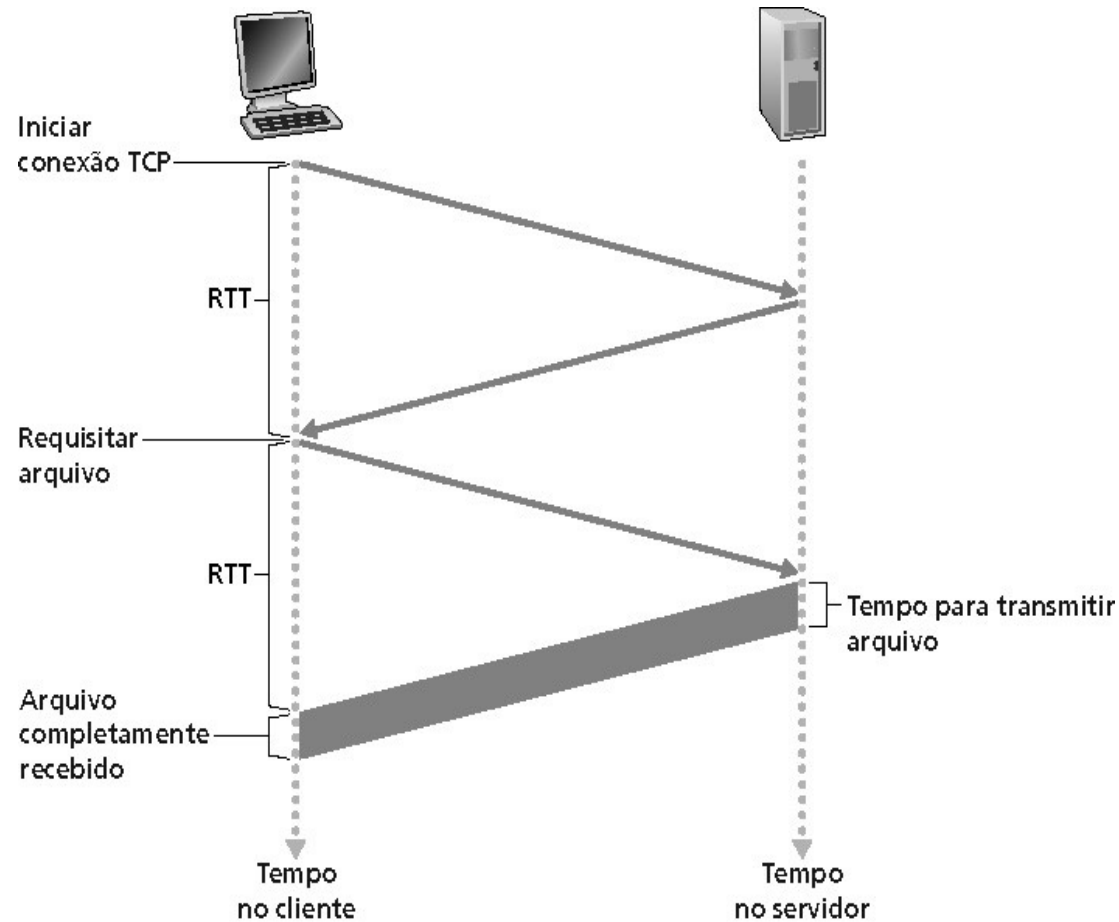


Visão geral do HTTP (2/2)

- Utiliza TCP:
 - Cliente inicia conexão TCP (cria socket) para o Servidor na porta 80.
 - Servidor aceita uma conexão TCP do Cliente.
 - Mensagens HTTP (mensagens do protocolo de camada de aplicação) são trocadas entre o browser (Cliente HTTP) e o Servidor Web (Servidor HTTP).
 - A conexão TCP é fechada ao final do envio dos objetos.
- HTTP é “stateless” (o oposto é chamado “stateful”).
 - O Servidor não mantém informação sobre os pedidos passados pelos Clientes. Se a sessão “cair”, precisa recomeçar o processo.

Modelagem do tempo de resposta

- Definição de RTT (Round Trip Time – tempo de ida e volta): tempo para enviar um pequeno pacote que vai do Cliente para o Servidor e retorna.
- Tempo de resposta:
 - Um RTT para iniciar a conexão TCP.
 - Um RTT para requisição HTTP e primeiros bytes da resposta HTTP para retorno.
 - Tempo de transmissão de arquivo:



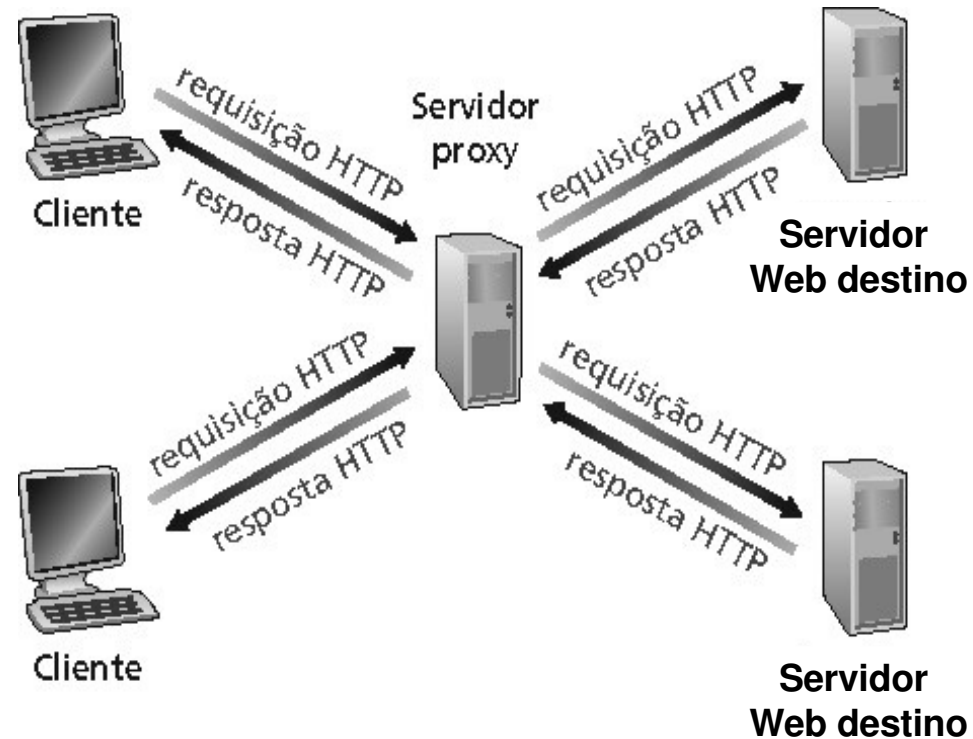
Total = $2RTT +$ tempo de transmissão do arquivo

Cookies: Estado usuário-Servidor

- São informações que permitem os sites monitorarem os usuários que o visitaram.
- A maioria dos Web sites utilizam cookies.
- Cookies possuem 4 componentes:
 - Linha de cabeçalho do cookie na mensagem HTTP de requisição.
 - Linha de cabeçalho de cookie na mensagem HTTP de resposta.
 - Arquivo de cookie mantido no Host do usuário e manipulado pelo browser do usuário associado a uma identificação (ID).
 - Restante das informações armazenadas no Banco de dados backend no Web site associada ao mesmo ID do cookie.

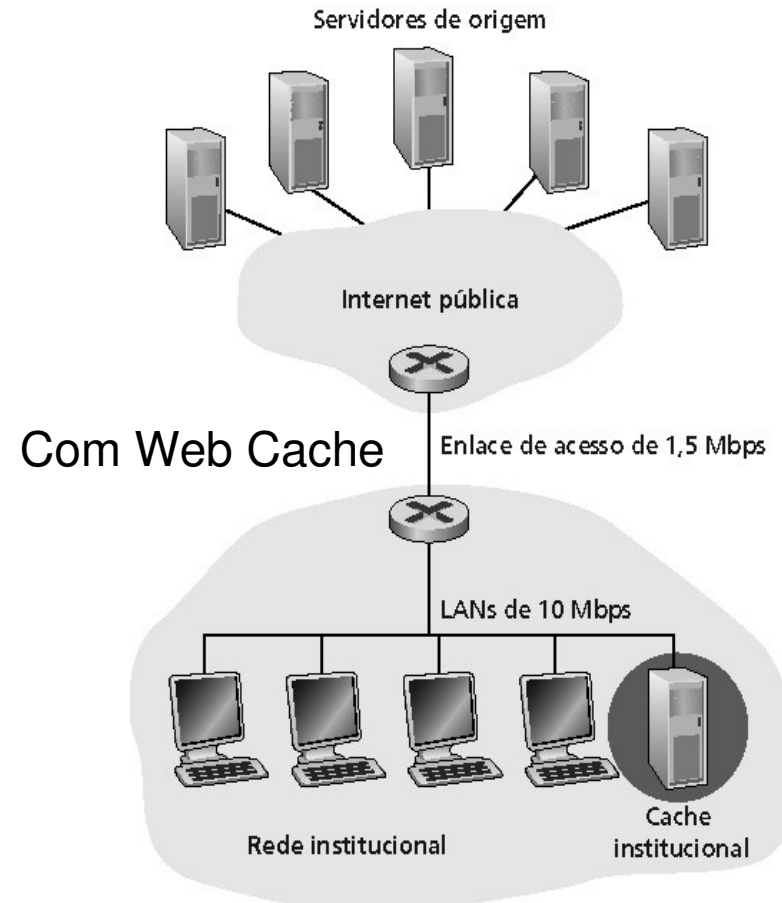
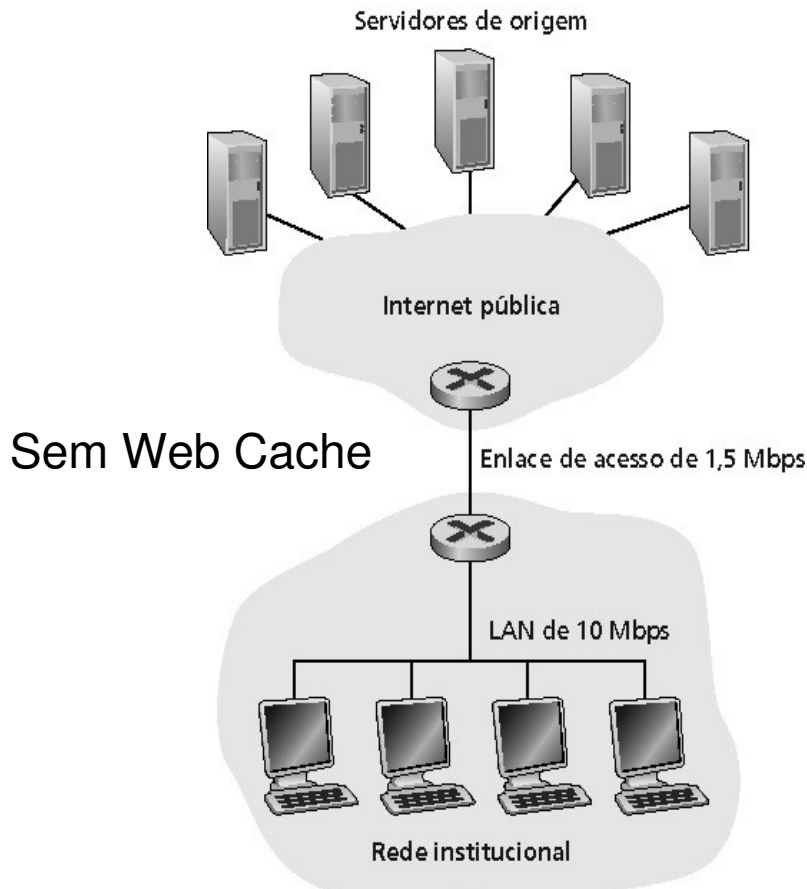
Web Caches (Proxy Server)

- Objetivo: atender a solicitação do Browser (Cliente) sem envolver o Servidor Web destino.
- Usuário configura o browser: acesso Web é feito por meio de um Servidor Proxy (Web Cache).
- Cliente envia todos os pedidos HTTP para o Web cache:
 - Se o objeto existe no Web cache: Web Cache retorna o objeto diretamente ao browser.
 - Ou o Web Cache solicita objeto do Servidor Web destino e então envia o objeto recebido ao Browser (Cliente).



Exemplo de caching

- Sem cache a página “caminha” pelo link de 1,5Mbps até o servidor destino (que pode estar do outro lado do mundo).
- Com cache, a página é recebida a 10Mbps na LAN (ou WAN próxima, no caso do cache no provedor Internet - ISP).



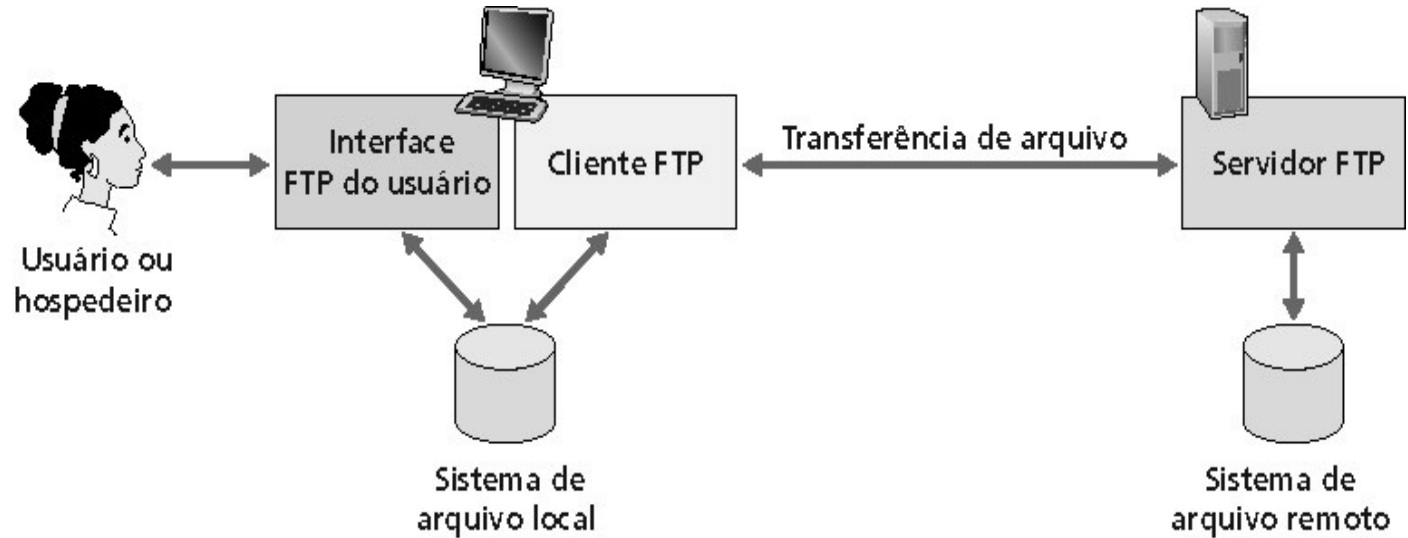
Por que usar Web Cache?

- Reduz o tempo de resposta para a requisição do Browser (Cliente). Se o Web Cache estiver na LAN, a velocidade de acesso é o da LAN.
- Reduz o tráfego na Internet e no enlace de acesso ao site desejado (Web Site destino), pois o Web Cache está mais “próximo” do Browser que do Servidor Web destino.
- A existência de Web Caches na Internet habilita os provedores “fracos” (com servidores de capacidade limitada e com links de acesso à Internet com velocidades limitadas) a efetivamente entregarem o conteúdo aos solicitantes, pois os objetos estão armazenados nos caches.

Comentário: Problema do Web Cache

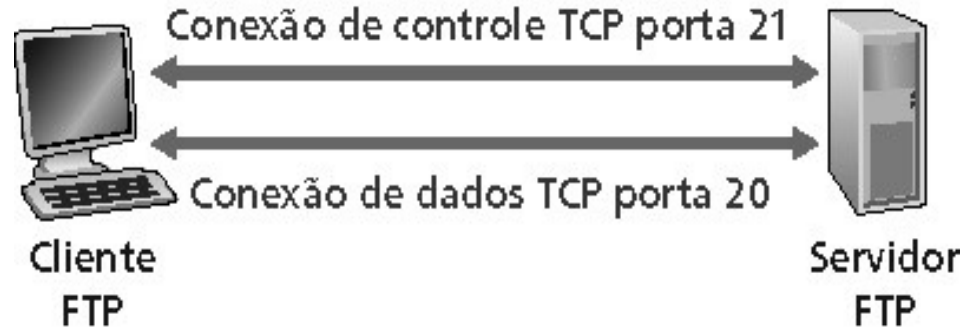
- Problema: O Servidor Web Cache pode entregar ao usuário, informação armazenada que esteja desatualizada.
- Solução: Usar msg de requisição GET Condicional.
- A mensagem Get Condicional inclui a informação da data e hora da última versão do objeto solicitado que foi recebida pelo Web Cache.
- O Servidor de destino verifica se a versão do objeto teve alteração após aquele instante:
 - Não houve alteração: Servidor Web destino envia mensagem “304 not modified” indicando que o objeto ainda é válido.
 - Houve alteração: Servidor Web destino envia o objeto atualizado que é armazenado no Web Cache substituindo o antigo e repassa o objeto ao browser do usuário solicitante.

FTP – File Transfer Protocol: protocolo de transferência de arquivos



- Transferência de arquivos de ou para o computador remoto (download ou upload respectivamente).
- Modelo Cliente Servidor.
 - Cliente: lado que inicia a transferência (seja de ou para o lado remoto).
 - Servidor: Host remoto.
- FTP: RFC 959.
- FTP Servidor: usa portas 20 e 21 do TCP.

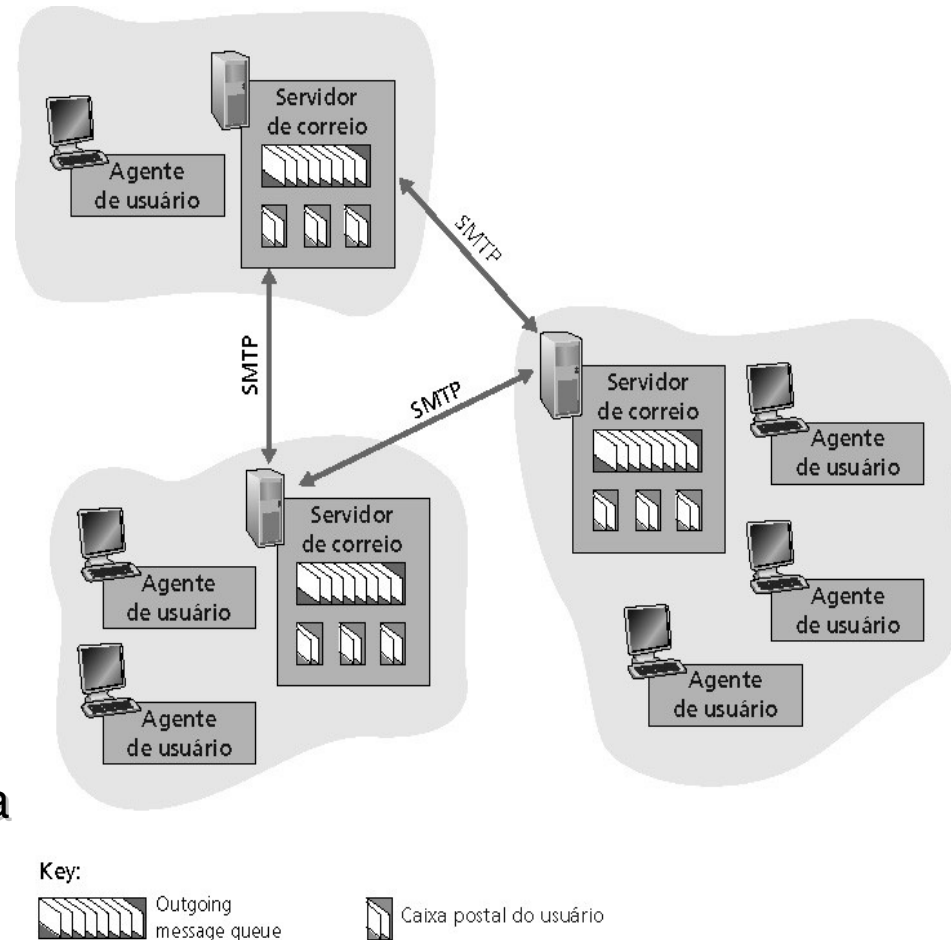
FTP: controle é separado das conexões de dados



- Cliente FTP contata o Servidor FTP na porta 21 especificando o TCP como protocolo de transporte.
- Cliente obtém autorização pela conexão de controle (porta 21).
- Cliente procura o diretório remoto no Servidor FTP enviando comandos pela conexão de controle.
- Quando o Servidor FTP recebe um comando para uma transferência de arquivo, ele abre uma conexão de dados TCP para o Cliente na porta 20.
- Após a transferência de um arquivo, o Servidor fecha a conexão de dados.
- Servidor abre uma segunda conexão de dados TCP para transferir outro arquivo, caso tenha sido solicitado pelo Cliente FTP.
- Conexão de controle: a conexão pela porta 21 é mantida enquanto o processo FTP estiver ativo no Cliente FTP.
- Servidor FTP mantém “estado”: diretório atual, autenticação anterior.

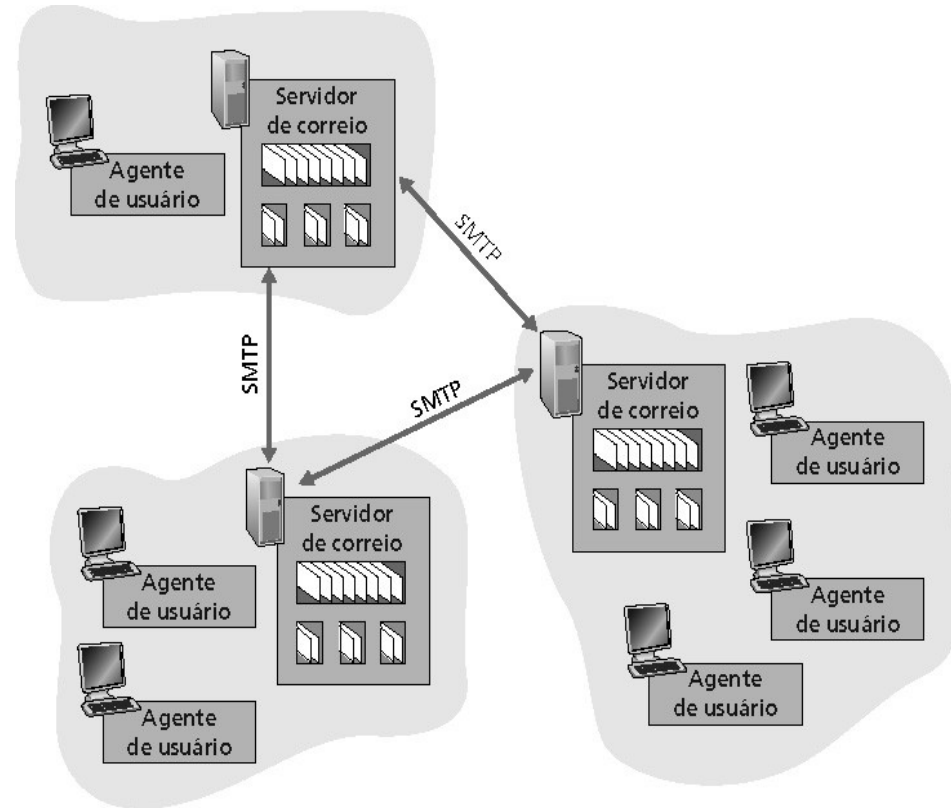
SMTP: Correio eletrônico (1/2)

- Três componentes principais:
 - Agentes de usuário.
 - Servidores de Correio.
 - Protocolo SMTP - Simple Mail Transfer Protocol.
- Agente de usuário:
 - Dá acesso do usuário ao seu Servidor de Correio (emails).
 - Composição, edição e leitura de mensagens de correio.
 - Mensagens de entrada e de saída são armazenadas no Servidor.
 - Ex.: Eudora, Outlook, Netscape Messenger.



SMTP: Correio eletrônico (2/2)

- Servidores de Correio:
 - Caixa postal contém mensagens que chegaram para o usuário.
 - Fila de mensagens contém as mensagens de correio a serem enviadas ao(s) outro(s) Servidor(es) de Correio.
 - Exemplo: MS-Exchange.
- Protocolo SMTP: permite aos Servidores de Correio trocarem mensagens entre si.
 - Cliente: Servidor de correio que envia.
 - Servidor: Servidor de correio que recebe.

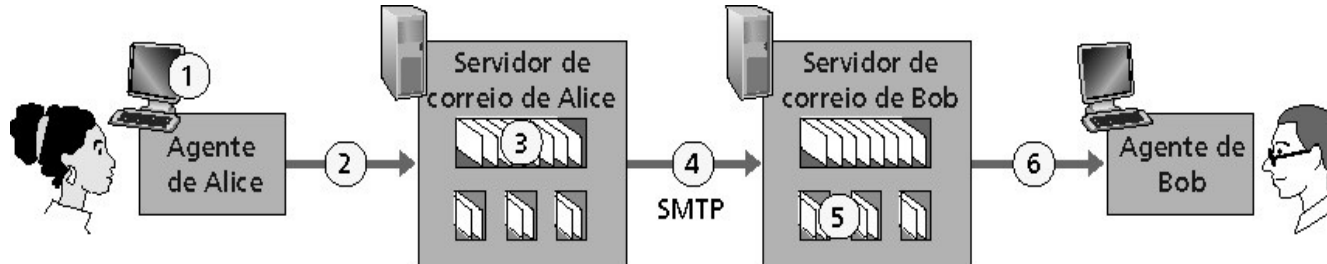


Key:

 Outgoing message queue

 Caixa postal do usuário

Cenário: Alice envia mensagem para Bob



Legenda:



Fila de mensagens



Caixa postal do usuário

- Alice usa o agente de usuário (UA) para compor a mensagem “para” bob@someschool.edu.
- O agente de usuário dela envia a mensagem para o seu Servidor de correio. A mensagem é colocada na fila de mensagens.
- O lado Cliente do SMTP abre uma conexão TCP com o Servidor de correio do Bob.
- O Cliente SMTP envia a mensagem de Alice pela conexão TCP.
- O Servidor de correio de Bob coloca a mensagem na caixa de correio de Bob.
- Bob invoca seu agente de usuário para ler a mensagem.

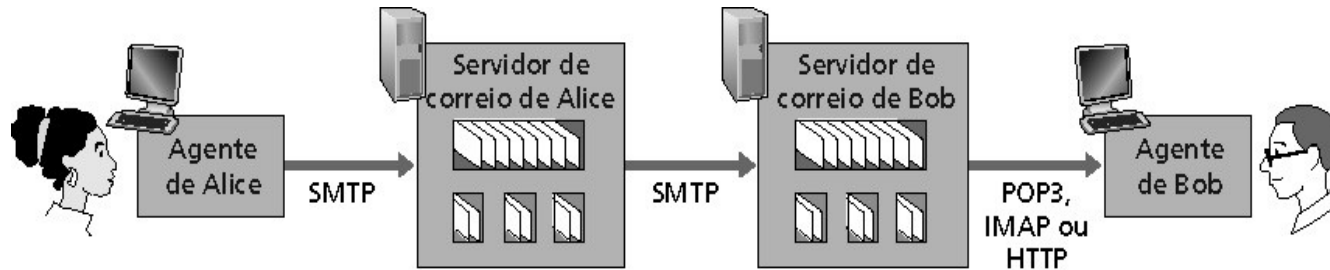
POP3 e IMAP

- POP3:
 - No modo “ler-e-apagar” (“download-and-delete”) os emails recebidos pelo usuário são deletados do Servidor de Correio. O usuário não pode reler o e-mail no Servidor de Correio.
 - No modo “ler-e-guardar” (“download-and-keep”), cópias das mensagens são mantidas no Servidor de Correio e podem ser lidas a qualquer tempo.
 - POP3 é stateless através das sessões.
- IMAP:
 - Mantém todas as mensagens em um lugar: o Servidor de Correio.
 - Permite que o usuário criar e organizar as mensagens em pastas dentro do Servidor de Correio.
 - IMAP mantém o estado do usuário através das sessões:
 - Nomes das pastas e mapeamentos entre os IDs da mensagem e o nome da pasta.

Webmail: Email pela Web

- O agente do usuário é um browser Web comum.
- O usuário se comunica com sua caixa postal remota via HTTP.
- Como no IMAP, o usuário pode criar e administrar suas pastas e mensagens armazenadas no Servidor de Correio.
- Muitas implementações de Webmail utilizam Servidor IMAP.

Protocolos de acesso ao correio



- SMTP: cuida da entrega e armazenamento no Servidor do destino.
- Protocolo de acesso: recupera mensagens do Servidor:
 - POP: Post Office Protocol [RFC 1939]:
 - Autorização (agente <-->Servidor) e download.
 - IMAP: Internet Mail Access Protocol [RFC 1730]:
 - Maiores recursos (mais complexo).
 - Manipulação de mensagens armazenadas no Servidor.
 - HTTP: Hotmail , Yahoo! Mail etc.

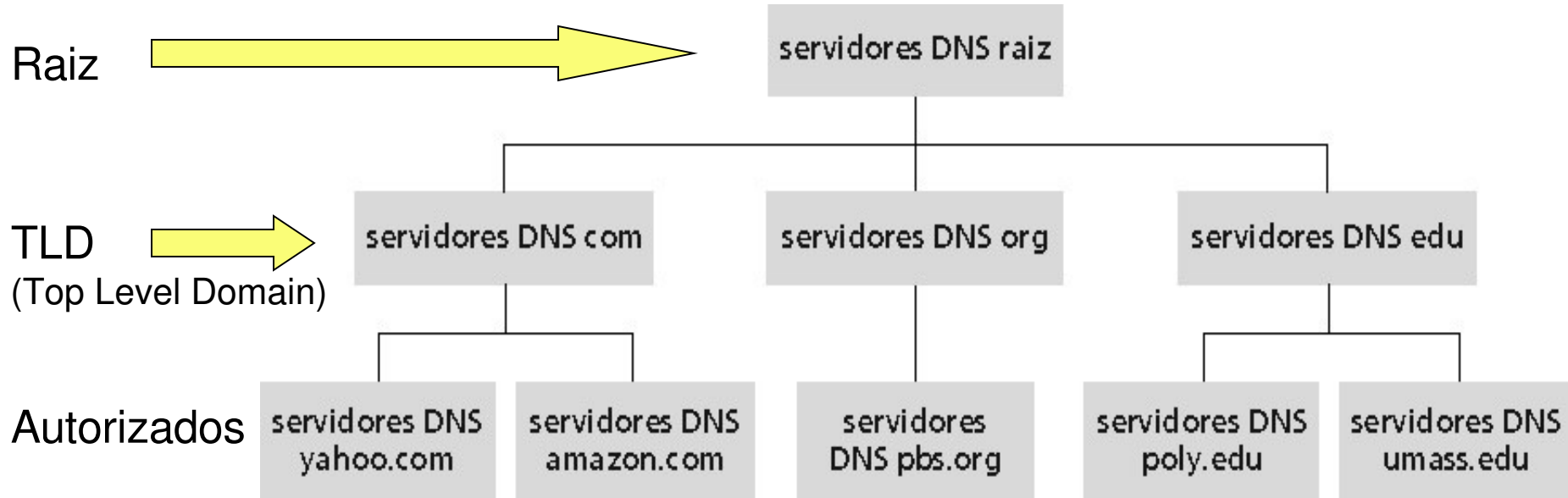
DNS - Domain Name System

- Identificação de Internet Hosts e roteadores:
 - Endereços IP (32 bits): usados para endereçar datagramas.
 - “nome”: ex.: gaia.cs.umass.edu - usados por humanos.
- Para relacionar nomes com endereços IP usamos DNS - Domain Name System.
- DNS – Domain Name System (também chamado de Domain Name Service):
 - Base de dados distribuída implementada de forma hierárquica com vários servidores de nomes que irão traduzir os nomes em endereço IP.
- É uma função interna da Internet, implementada como protocolo da camada de aplicação.

Serviços prestados pelo DNS

- Tradução do nome do Host em endereço IP.
- Host “aliasing” (“apelidos”): alguns nomes de servidores não são fáceis de se lembrar então o DNS atribui um nome mais “fácil” de lembrar:
 - Nomes canônicos (nome completo) e alias (apelido): Exemplo: nome canônico “relay.west-coast.enterprise.com” tem o apelido de “enterprise.com”.
 - Mail server aliasing: Exemplo: “relay.west-coast.hotmail.com” tem o apelido “hotmail.com”.
- Distribuição de carga:
 - Servidores Web replicados: um nome canônico estará associado a um conjunto de endereços IP. O DNS fornece a lista de todos os IPs desses servidores replicados para o solicitante que informou o nome canônico.
 - O DNS, a cada consulta desse nome canônico, faz um rodízio da ordem dos endereços IP.
 - Arquitetura hierárquica evita um único ponto de falha e otimiza o tráfego na Internet.

Base de dados distribuída, hierárquica



- Cliente quer o IP para `www.amazon.com`:
 - Cliente consulta um Servidor de raiz para encontrar o Servidor DNS “com”.
 - Cliente consulta o Servidor DNS “com” para obter o Servidor DNS “amazon.com”.
 - Cliente consulta o Servidor DNS “amazon.com” para obter o endereço IP para “www.amazon.com”.

DNS: Servidores de nomes raiz

- São contatados pelos Servidores de nomes locais que não podem resolver um nome.
- Servidores de nomes raiz:
 - Buscam Servidores de nomes autorizados se o mapeamento do nome não for conhecido.
 - Conseguem o mapeamento.
 - Retornam o mapeamento para o servidor de nomes local.
- Em fevereiro/2004 eram 13 servidores de nomes raiz no mundo (figura).



Servidores TLD e Autorizados

- Servidores top-level domain (TLD):
 - São responsáveis pelos domínios com, org, net, edu etc e todos os domínios top-level nacionais uk, fr, ca, jp.
- Servidores DNS Autorizados:
 - São Servidores DNS de organizações que provêm nome de Host autorizado para mapeamentos IP para Servidores de organizações (ex.: Web e mail).
- Podem ser mantidos por uma organização ou provedor de serviços.

Servidor de nomes local

- Não pertence estritamente a uma hierarquia.
- Cada ISP (ISP residencial, companhia, universidade) possui um
 - Também chamado de “Servidor de nomes default”.
- Quando um Host faz uma pergunta a um DNS, a pergunta é enviada para seu Servidor DNS local.
 - Age como um proxy, encaminhando as perguntas para dentro da hierarquia.
- Uma vez que um Servidor de nomes apreende um mapeamento, ele armazena o mapeamento num registro do tipo cache.
 - Registro do cache tornam-se obsoletos (desaparecem) depois de um certo tempo.
 - Servidores TLD são tipicamente armazenados em cache nos Servidores de nome locais.

Exemplo do processo de tradução no DNS

- O Host em cis.poly.edu quer o endereço IP para gaia.cs.umass.edu.
- Seguir a sequência numérica dos passos desenvolvidos no processo de tradução DNS.

